



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах

Лабораторна робота №3
Архітектура комп'ютера
«ДОСЛІДЖЕННЯ МЕХАНІЗМІВ АДРЕСАЦІЇ АРХІТЕКТУРИ IA-32 (X86)
У REAL ADDRESS MODE»

Виконали:
студенти групи ІТ-01:

Перевірив:
Бердник Ю. М.

[Бардін В.Д.](#)
[Задніпрянець А.А.](#)
[Куркін О.О.](#)

Дата здачі 10.02.2021

Захищено з балом _____

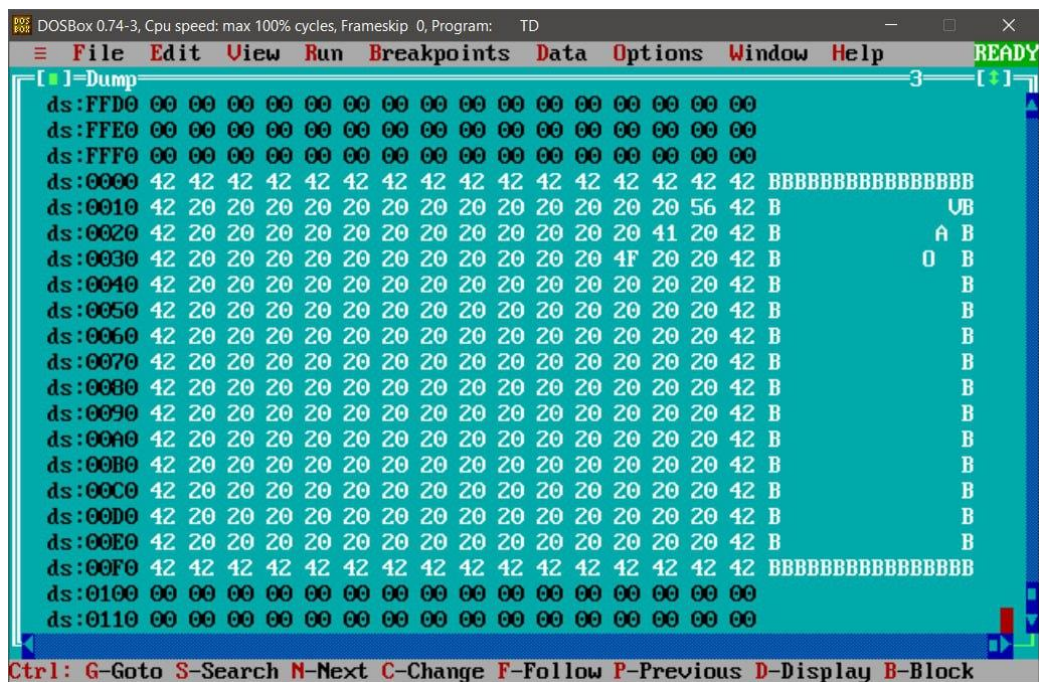
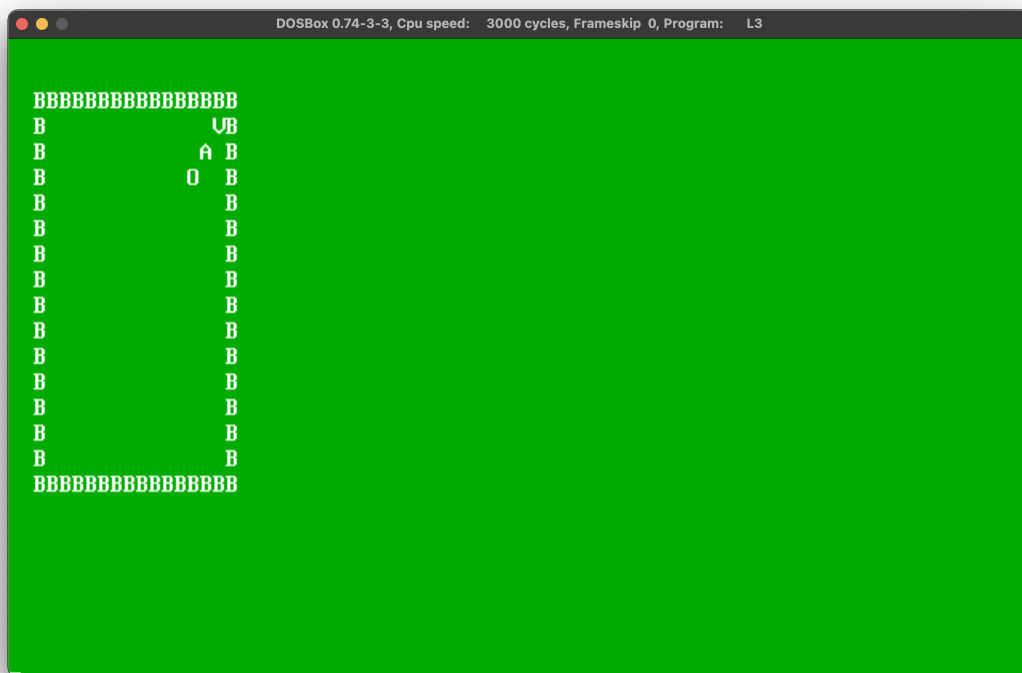
Вихідний код:

```
1 TITLE Lab-3
2 ;-----
3 ; ЛР №3
4 ;-----
5 ; Архітектура комп'ютера.
6 ; Завдання: Основи розробки і налагодження
7 ; ВУЗ: НТУУ "КПІ"
8 ; Факультет: ФІОТ
9 ; Курс: 1
10 ; Група: IT-01
11 ;-----
12 ; Автор: Bardin Vladislav,
13 ; Zadniprianets Artur,
14 ; Kurkin Oleksii
15 ; Дата: 23/02/2021
16 ;-----I.ЗАГОЛОВОК ПРОГРАМИ-----
17 IDEAL
18 MODEL SMALL
19 STACK 512
20 ;-----II.МАКРОСИ-----
21 ; Макрос для ініціалізації
22 MACRO M_Init ; Початок макросу
23 mov ax,@data
24 mov ds,ax ; ds = data segment
25 mov ax,0B800h
26 mov es,ax ; es = text VRAM segment for direct VRAM writes
27 ENDM M_Init ; Кінець макросу
28
29 ;-----III.ПОЧАТОК СЕГМЕНТУ ДАНИХ-----
30 DATASEG
31
32 ; масив, який буде записано в пам'ять, а потім з неї надруковано
33 matrix db 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h
34 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 'V', 42h
35 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 'A', 20h, 42h
36 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 'O', 20h, 42h
37 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
38 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
39 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
40 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
41 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
42 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
43 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
44 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
45 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
46 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
47 db 42h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 20h, 42h
48 db 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h, 42h
49
50 BOARD_SIZE_X EQU 16 ; Розмір масиву по осі 0x
51 BOARD_SIZE_Y EQU 16 ; Розмір масиву по осі 0y
52 START_X EQU 2 ; Координата X верхнього лівого кута матриці
53 START_Y EQU 2 ; Координата Y верхнього лівого кута матриці
```

```

54 ;-----VI. ПОЧАТОК СЕГМЕНТУ КОДУ-----
55 CODESEG
56
57 Start:
58     M_Init
59
60     mov ah, [matrix]           ; Запис масиву у старші біти регістру ah
61
62     call clear_screen          ; Виклик процедури, яка зафарбує екран у зелений колір
63     call draw_board            ; Виклик процедури, яка намалює екран
64
65     xor ah,ah                  ; Очищуємо регістр ah і очікуємо натискання клавіші для того, щоб
    продовжити виконання роботи
66     int 16h
67
68     mov ax,4C00h               ; Завершуємо програму
69     int 21h
70
71
72 ;-----Очистка екрану від зайвого тексту-----
73 ; Вхідні параметри - немає
74 ; Вихідні параметри - немає
75 ;-----
76
77     PROC clear_screen
78     xor di,di                  ; Очищення di, перехід до адреси B800:0000
79     mov ax, ' ' + 2Fh*256      ; Заповнення екрана пробілами на зеленому фоні
    mov cx,80*25                ; Кіл-сть символів, яку нам потрібно замалювати (cols * rows). DOSBox
    width 80 cols and height 25 rows
80     rep stosw                  ; Заповнення di <- ax, доки cx > 0
81     ret
82     ENDP
83
84 ;-----Друк масиву на екран-----
85 ; Вхідні параметри - немає
86 ; Вихідні параметри - немає
87 ;-----
88
89     PROC draw_board
90     mov si,offset matrix       ; Встановлюємо у регістр si посилання на перший елемент масиву
91
92     ; Зміщення точки друку за формулою (80 * y + x) * 2
93     ; 2 bytes per char (char+color), 80 chars (160B) per line
94     mov di,(START_Y*80 + START_X)*2
95     ; Друк BOARD_SIZE_Y рядків
96     mov dx,BOARD_SIZE_Y
97     board_line_loop:
98     ; Друк BOARD_SIZE_X символів
99     mov cx,BOARD_SIZE_X
100    board_char_loop:
101    lodsb                       ; Переміщення в регістрі si + 1
102    stosw                       ; Запис символу + його кольору до відеопам'яті es:di, переміщення di +2
103    dec cx
104    jnz board_char_loop ; Ітерація по кожному символу, який потрібно надрукувати
105    ; Перенесення вказівника відеопам'яті в початок наступного рядка
106    add di,(80-BOARD_SIZE_X)*2 ; Доповнення символами, щоб перейти на наступний рядок
107    dec dx
108    jnz board_line_loop ; Прохід по кожному рядку, який необхідно намалювати
109    ret
110    ENDP
111 end Start

```



Результати дослідження: Було створено двовимірний масив, що складається з елементів в один байт, має розмір 16x16 та записали на діагональ масиву ініціали студентів бригади.

Висновок: У ході лабораторної роботи ознайомилися з видами адресації та створили двовимірний масив.